



TUGAS AKHIR - KI091391

PENERAPAN SISTEM PENCATATAN DATA LOKASI PADA WEB DAN PERANGKAT BERGERAK (ANDROID)

I PUTU SUDHYANA MECHA
NRP 5109100100

Dosen Pembimbing
Ary Mazharuddin S., S.Kom, M.Comp.Sc.

JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2014



UNDERGRADUATE THESIS - KI091391

IMPLEMENTATION OF RECORDING LOCATION DATASYSTEM USING WEB AND MOBILE DEVICE (ANDROID)

**I PUTU SUDHYANA MECHA
NRP 5109100100**

**Supervisor
Ary Mazharuddin S., S.Kom, M.Comp.Sc.**

**INFORMATICS DEPARTMENT
Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2014**

PENERAPAN SISTEM PENCATATAN DATA LOKASI PADA WEB DAN PERANGKAT BERGERAK (ANDROID)

TUGAS AKHIR

**Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Komputasi Berbasis Jaringan
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember**

Oleh :

I PUTU SUDHYANA MECHA

NRP : 5109100100

Disetujui oleh Dosen Pembimbing Tugas Akhir :

Ary Mazharuddin S., S.Kom, M. Comp. Sc.
NIP: 198106202005011003




.....
(Pembimbing)

**SURABAYA
JUNI 2014**

PENERAPAN SISTEM PENCATATAN DATA LOKASI PADA WEB DAN PERANGKAT BERGERAK (ANDROID)

Nama Mahasiswa : I PUTU SUDHYANA MECHA
NRP : 5109 100 100
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing : Ary Mazharuddin S., S.Kom,
M.Comp.Sc.

Abstrak

Pencatatan data lokasi oleh petugas Perusahaan Daerah Air Minum (PDAM) merupakan salah satu contoh kasus yang berkaitan dengan kehidupan sehari-hari. Petugas PDAM akan mencatat setiap detil lokasi beserta angka meter air yang tertera pada setiap lokasi rumah yang dikunjunginya. Pada mulanya, proses pencatatan ini berjalan sebagaimana mestinya. Akan tetapi, timbul permasalahan baru dimana data yang diperoleh dari hasil pencatatan data lokasi tersebut tidak sesuai dengan kenyataan sesungguhnya di lapangan. Bisa saja data yang diperoleh tersebut direkayasa ataupun tidak sesuai dengan kenyataannya di lapangan.

Diperlukan adanya suatu sistem yang memungkinkan pencatatan data lokasi dilakukan secara *online*. Dengan perangkat bergerak (*android*) yang terhubung dengan perangkat *web*, maka *user* dan *administrator* bisa menjalankan perannya dengan berkesinambungan. *Administrator* disini berperan sebagai bagian *monitoring*, yang bertugas melihat data lokasi yang nantinya akan dilaporkan oleh *user*. Sedangkan *user*, dalam kasus ini petugas pencatat PDAM, merupakan pengguna perangkat bergerak yang akan memasukkan dan mengirimkan data lokasi, agar *administrator* bisa memantau setiap laporan *user* yang masuk.

Sesuai dengan gambaran di atas, maka sistem akan dibangun dalam dua *platform* utama, yaitu *web* dan *android*. Pada *web* akan dibangun 5 fitur utama, yaitu *menu login web*, *menu utama administrator*, *menu entry administrator*, *menu peta administrator*, dan *menu peta user*. Sedangkan pada *android* akan dibangun 2 fitur utama, yaitu *menu login android* dan *menu entry user*.

Kata Kunci : data lokasi, *web*, *android*, sistem pencatat

IMPLEMENTATION OF RECORDING LOCATION DATA SYSTEM USING WEB AND MOBILE DEVICE (ANDROID)

Name : I PUTU SUDHYANA MECHA
NRP : 5109 100 100
Major : Informatics Department Faculty of IT – ITS
Supervisor : Ary Mazharuddin S., S.Kom, M.Comp.Sc.

Abstract

Recording *location data* by officers taps (PDAM) is one example of a case related to everyday life. Officers taps will record every detail of the sites and number of water meters at each location listed homes he visited. At first, the recording process is running as it should be. However, new problems arise where the data obtained from the recording location data is not in accordance with the actual reality on the ground. It could be data obtained was manipulated, or not in accordance with the reality in the daily life.

It is necessary to have a system that allows the recording of location data is done online. With mobile devices (*android*) which is connected to the *web*, so users and administrators can perform its role in sustainable. Administrators play a role here as part of the monitoring, which is in charge of seeing that location data will be reported by the user. While the user, in this case the registrator taps, who is a mobile device user will enter and transmit location data, so that the administrator can monitor any incoming user reports.

Based on the description above, then the system will be built in two major platforms, such as web and android. On the web will be built 5 main features, namely a web login menu, the administrator main menu, the administrator menu entry, the administrator maps menu, and the menu map user. On the android will be built 2 main features, namely the login menu and the menu entry android user.

Keywords : *location data, web, android, recording system*

KATA PENGANTAR

Puja dan puji syukur Penulis panjatkan ke hadapan Tuhan Yang Maha Kuasa, karena atas restu-Nya lah Penulis dapat menyelesaikan Tugas Akhir yang berjudul “**Penerapan Sistem Pencatatan Data Lokasi Pada Web Dan Perangkat Bergerak (Android)**” ini.

Dalam pelaksanaan dan pembuatan Tugas Akhir ini tentunya sangat banyak bantuan-bantuan serta inspirasi yang Penulis terima dari berbagai pihak, tanpa mengurangi rasa hormat penulis ingin mengucapkan terima kasih kepada :

1. Sang Maha Pencipta, Pemelihara, dan Pelebur, Tuhan Yang Maha Kuasa atas restu-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik.
2. Kedua orang tua Penulis, Bapak I Ketut Gede Wirawan dan Ibu Ni Gusti Nyoman Suci Murni, adik Made Sandhyana Angga dan I Nyoman Prayana Trisna yang telah memberikan banyak sekali dukungan moral, spiritual dan material, semangat, perhatian, selalu setia dan sabar dalam menghadapi curhatan dari penulis, serta selalu memberikan doa yang tiada habisnya yang dipanjatkan untuk penulis.
3. Bapak Ary Mazharuddin Shiddiqi, S.Kom, M.Comp.Sc., selaku dosen pembimbing, yang telah memberikan kepercayaan, dukungan, bimbingan, nasehat, perhatian, serta semua yang telah diberikan kepada penulis.
4. Ir. Bapak Rully Soelaiman, M.Kom., selaku Bapak, motivator, dan juga inspirator penulis, sehingga mampu untuk terus semangat dalam menjalankan dan menyelesaikan tugas akhir.
5. Ibu Nanik Suciati, S.Kom., M.Kom., Dr.Eng. selaku ketua jurusan Teknik Informatika ITS, Bapak Waskhito Wibisono, S.Kom., M.Eng. selaku dosen wali, Bapak Dr. Eng. Radityo Anggoro, S.Kom, M.Sc., selaku koordinator

TA, dan segenap Bapak/Ibu dosen Teknik Informatika yang telah memberikan ilmunya kepada Penulis. Segenap staf Tata Usaha yang telah memberikan segala bantuan dan kemudahan kepada Penulis selama menjalani kuliah di Teknik Informatika ITS.

6. Wira, Andi, Suar, Gosha, Goblet, Candra, dan segenap anak kost J2 atas *support* dan bantuannya dalam pengerjaan Tugas Akhir ini.
7. Teman-teman SMA Negeri 3 Denpasar angkatan XXX, khususnya kelas IPA 4 yang sampai saat ini masih setia memberikan dukungan dan semangatnya untuk Penulis.
8. Seluruh teman Teknik Informatika ITS angkatan 2009 dan Himpunan Mahasiswa Teknik Computer-Informatika C-19, terima kasih atas rasa kekeluargaan dan solidaritas yang tinggi kepada penulis. Semoga kita semua menjadi orang yang sukses dalam bidangnya masing-masing.
9. Bapak Dharmaputra selaku pemberi ide/gagasan mengenai Tugas Akhir ini.
10. Juga tidak lupa kepada semua pihak yang belum sempat disebutkan satu per satu yang telah membantu terselesaikannya Tugas Akhir ini.

Penulis mengharapkan adanya saran dan kritik yang membangun dari pembaca, sehingga memperlancar Tugas Akhir ini agar dapat menjadi manfaat bagi masyarakat.

Surabaya, Juni 2014

Penulis

DAFTAR ISI

Abstrak	ix
Abstract	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
DAFTAR KODE IMPLEMENTASI	xxiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan	2
1.3 Rumusan Masalah.....	2
1.4 Batasan Masalah	3
1.5 Metodologi.....	3
1.5.1 Studi Kepustakaan	3
1.5.2 Perancangan Sistem	3
1.5.3 Implementasi.....	3
1.5.4 Uji Coba dan Evaluasi	4
1.5.5 Penyusunan Laporan Tugas Akhir.....	4
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	7
2.1 Android	7
2.1.1 JSONParser.....	7
2.1.2 Shared Preference	7
2.1.3 AsyncTask.....	8
2.2 PHP	8
2.3 JavaScript (Google Maps API)	9
2.4 Web Service	10
BAB III PERANCANGAN PERANGKAT LUNAK	11
3.1 Deskripsi Umum	11
3.2 Arsitektur Aplikasi.....	11
3.3 Skenario Use Case	12
3.4 Fitur Aplikasi	14
3.5 Proses Utama	15

3.5.1	Proses Pada Web.....	15
3.5.2	Proses Pada Android.....	16
3.6	Diagram Alir	16
3.6.1	Diagram Alir Masuk Ke Dalam Sistem (Login Web).....	16
3.6.2	Diagram Alir Melihat Data Last Update Setiap User.....	17
3.6.3	Diagram Alir Melihat Keterangan Lokasi Seluruh User.....	18
3.6.4	Diagram Alir Memasukkan Data Lokasi User.....	19
3.6.5	Diagram Alir Masuk Ke Dalam Sistem (Login Android).....	20
3.6.6	Diagram Alir Mengirimkan Data Lokasi Ke Database.....	21
3.6.7	Diagram Alir Melihat Keterangan Lokasi Setiap User.....	22
3.7	Rancangan Antarmuka.....	23
3.7.1	Antarmuka Web.....	24
3.7.2	Antarmuka Android	27
BAB IV IMPLEMENTASI PERANGKAT LUNAK.....		29
4.1	Lingkungan Implementasi	29
4.1.1	Lingkungan Perangkat Lunak	29
4.1.2	Lingkungan Perangkat Keras.....	30
4.2	Implementasi Perangkat Lunak.....	30
4.2.1	Implementasi Web	30
4.2.2	Implementasi Android	35
BAB V UJI COBA DAN EVALUASI		39
5.1	Lingkungan Uji Coba.....	39
5.2	Skenario Uji Coba.....	40
5.2.1	Uji Coba Fungsionalitas.....	40
5.2.2	Uji Coba Performa	51
5.2.3	Uji Coba Kompatibilitas	53
5.3	Evaluasi Uji Coba	54
BAB VI PENUTUP		55
6.1	Kesimpulan	55

6.2	Saran	56
7.	DAFTAR PUSTAKA	57
8.	LAMPIRAN	59
9.	BIODATA PENULIS	69

DAFTAR TABEL

Tabel 3.1 Tabel Use Case.....	13
Tabel 5.2 Uji Coba Masuk Ke Dalam Sistem	42
Tabel 5.3 Uji Coba Melihat Data Last Update Setiap User	43
Tabel 5.4 Uji Coba Melihat Keterangan Lokasi Seluruh User.....	44
Tabel 5.5 Uji Coba Memasukkan Data Lokasi User	46
Tabel 5.6 Uji Coba Masuk Ke Dalam Sistem (Login Android) ..	47
Tabel 5.7 Uji Coba Mengirimkan Kode Lokasi Ke Database	49
Tabel 5.8 Uji Coba Melihat Keterangan Lokasi Setiap User	50
Tabel 5.9 Uji Coba Performa Pada Web	51
Tabel 5.10 Uji Coba Performa Pada Android	53
Tabel 5.11 Uji Coba Kompatibilitas.....	53
Tabel 5.12 Evaluasi Uji Coba Fungsionalitas	54

DAFTAR GAMBAR

Gambar 2.1 Halaman Google Maps API.....	9
Gambar 3.2 Arsitektur Aplikasi	12
Gambar 3.3 Use Case Diagram	13
Gambar 3.4 Diagram Alir Masuk Ke Dalam Sistem.....	17
Gambar 3.5 Diagram Alir Melihat Data Last Update Setiap User.....	18
Gambar 3.6 Diagram Alir Melihat Keterangan Lokasi Seluruh User	19
Gambar 3.7 Diagram Alir Memasukkan Data Lokasi User	20
Gambar 3.8 Diagram Alir Masuk ke Dalam Sistem (Login Android)	21
Gambar 3.9 Diagram Alir Mengirimkan Data Lokasi ke Database	22
Gambar 3.10 Diagram Alir Melihat Keterangan Lokasi Setiap User	23
Gambar 3.11 Antarmuka Halaman Beranda Login	24
Gambar 3.12 Antarmuka Halaman Menu Utama Administrator	25
Gambar 3.13 Antarmuka Halaman Menu Peta Administrator	25
Gambar 3.14 Antarmuka Halaman Menu Entry Administrator ..	26
Gambar 3.15 Antarmuka Halaman Menu Peta User	26
Gambar 3.16 Antarmuka Halaman Beranda Login	27
Gambar 3.17 Antarmuka Halaman Menu Utama User	27
Gambar 4.18 Halaman Login Web.....	31
Gambar 4.19 Pseudocode Halaman Login Web.....	31
Gambar 4.20 Halaman Menu Utama Administrator	32
Gambar 4.21 Pseudocode Halaman Menu Utama Administrator	32
Gambar 4.22 Halaman Menu Peta Administrator	33
Gambar 4.23 Pseudocode Halaman Menu Peta Administrator ...	33
Gambar 4.24 Halaman Menu Entry Administrator	34
Gambar 4.25 Pseudocode Halaman Menu Entry Administrator	34
Gambar 4.26 Halaman Menu Peta User	35
Gambar 4.27 Pseudocode Halaman Menu Peta User	35
Gambar 4.28 Halaman Login Android.....	36

Gambar 4.29 Pseudocode Halaman Menu Login User	36
Gambar 4.30 Halaman Menu Entry User	37
Gambar 4.31 Pseudocode Halaman Menu Entry User.....	38
Gambar 5.32 Uji Coba Masuk Ke Dalam Sistem (Login Web) ..	41
Gambar 5.33 Berhasil Masuk Ke Dalam Sistem (Administrator).....	41
Gambar 5.34 Berhasil Masuk Ke Dalam Sistem (User).....	42
Gambar 5.35 Uji Coba Melihat Data Last Update Setiap User...	43
Gambar 5.36 Uji Coba Melihat Keterangan Lokasi Seluruh User	44
Gambar 5.37 Uji Coba Memasukkan Data Lokasi User	45
Gambar 5.38 Berhasil Memasukkan Data Lokasi User	46
Gambar 5.39 Uji Coba Masuk Ke Dalam Sistem (Login Android)	47
Gambar 5.40 Uji Coba Mengirimkan Data Lokasi Ke Database	48
Gambar 5.41 Uji Coba Melihat Keterangan Lokasi Setiap User	50

BAB I

PENDAHULUAN

Pada bab ini akan dijelaskan beberapa hal dasar yang berhubungan dalam pembuatan Tugas Akhir antara lain latar belakang, perumusan masalah, tujuan, dan manfaat pembuatan Tugas Akhir serta metodologi dan sistematika pembuatan buku ini. Dari uraian dibawah ini, diharapkan dapat memudahkan mendapatkan gambaran Tugas Akhir secara umum dengan baik.

1.1 Latar Belakang

Pencatatan angka meter air oleh petugas Perusahaan Daerah Air Minum (PDAM) telah dilakukan selama beberapa tahun belakangan ini. Setiap bulannya, petugas melaporkan data dari penduduk setempat ke kantor pusat (kota). Data yang dilaporkan berupa nama penduduk pengguna PDAM beserta angka meter air terakhir di bulan tersebut. Untuk memperoleh data ini, petugas melakukan pencatatan secara *manual*, yakni dari satu rumah ke rumah yang lain, dan dicatat pada lembar tertentu, sesuai dengan pembagian tugas yang telah diberikan.

Sistem pencatatan ini berlangsung dengan baik, namun bukan berarti telah berjalan dengan efektif. Inilah yang sebenarnya menjadi fokus utama permasalahan yang akan penulis coba selesaikan. Masalah utamanya bukanlah terletak pada metode pencatatannya yang *manual*, namun lebih pada *validitas* datanya, dalam hal ini angka meter air. Tidak ada bukti nyata yang bisa menunjukkan bahwa petugas PDAM telah mencatat angka meter air penduduk secara langsung dan benar adanya. Dibutuhkan sesuatu yang bersifat teknis untuk bisa menunjukkan bahwa data yang dilaporkan oleh petugas merupakan data yang benar/*valid*.

Pencatatan data lokasi merupakan teknologi kekinian yang memungkinkan penulis untuk mengetahui kebenaran lokasi dari seseorang. Dengan adanya *mobile device (mobile phone)*, penulis bisa mengetahui keberadaan orang tersebut berdasarkan lokasi

dimana terakhir kali melakukan pelaporan (*report*). Selain itu, penulis juga bisa menampilkan lokasi yang harus dikunjungi *user* dan menampilkannya ke dalam peta *digital (maps)*.

Aplikasi ini mengadaptasi arsitektur *client to web*, dimana aplikasi ini mampu mencatat koordinat posisi dari *user*, dalam hal ini petugas PDAM dan melaporkannya langsung ke kantor pusat (kota). Jadi, kantor pusat bisa mengetahui secara langsung apakah petugas benar-benar datang ke rumah penduduk untuk mendata meter air atau tidak. Tentunya, aplikasi ini diharapkan mampu mempermudah pencatatan data lokasi dari petugas PDAM, dimana setiap petugas mempunyai banyak lokasi yang harus dikunjungi.

1.2 Tujuan

Tujuan dari pengerjaan Tugas Akhir ini adalah :

1. Memberikan gambaran yang tepat dan jelas kepada pengguna aplikasi mengenai sistem pencatat data lokasi ini.
2. Memberikan informasi yang tepat dan jelas mengenai lokasi terakhir kali (*last updated*) pengguna *client (current location)* kepada pengguna *web* (dalam hal ini *administrator*).

1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, terdapat beberapa permasalahan yang dapat dirumuskan sebagai berikut :

1. Bagaimana mengimplementasikan penggunaan *Android* pada *client (mobile)*.
2. Bagaimana mengimplementasikan penggunaan *maps (Google Maps)* yang akan di-intregasikan pada *web (PHP)*.
3. Bagaimana mengimplementasikan penggunaan *web* yang akan digunakan untuk menampilkan lokasi *user*.

1.4 Batasan Masalah

Adapun batasan masalah pada pengerjaan Tugas Akhir ini adalah :

1. Arsitektur perangkat lunak dari aplikasi pencatat lokasi ini berupa *client to web*
2. Aplikasi *client* dijalankan pada *mobile (Android)*
3. Aplikasi *web* dijalankan pada web *PHP*
4. *Maps* yang diintegrasikan dalam *web (PHP)* adalah *Google Maps*
5. Lokasi *maps* yang ditampilkan dalam *web (PHP)* hanya satu kota tertentu saja

1.5 Metodologi

1.5.1 Studi Kepustakaan

Tahap ini merupakan tahap lebih dalam mengenai proposal yang telah dirancang sebelumnya. Pada tahap ini, akan dipelajari beberapa sumber-sumber yang berkaitan dengan sistem pencatatan data lokasi, diantaranya adalah :

1. *Android*
2. *PHP*
3. *JavaScript*
4. *Web Service*

1.5.2 Perancangan Sistem

Tahap perancangansistem ini merupakan tahap yang menjelaskan dan menggambarkan bagaimana sistem perangkat lunak bekerja berdasarkan studi kepustakaan yang telah dilakukan. Tahap ini mencakup bagaimana rancangan arsitektur, alur kerja atau proses bisnis, serta rancangan antarmuka perangkat lunak sistem. Hal ini dilakukan untuk mengetahui gambaran sistem seperti apa yang akan diimplementasikan pada perangkat lunak.

1.5.3 Implementasi

Tahap ini merupakan tahap yang menjelaskan seperti apa penerapan yang dilakukan pada perangkat lunak berdasarkan desain sistem yang telah dirancang. Selain itu, tahap ini juga

menjelaskan apa saja yang digunakan untuk membangun sistem, dimana bisa dilihat pada poin-poin berikut ini :

1. *Web* dibangun menggunakan *PHP* yang diintegrasikan dengan *Google Maps*.
2. *Client* dibangun menggunakan *Android Developmet Tools* yang diintegrasikan dengan *Android Phone SDK*.
3. *Database* dibangun menggunakan *MySQL* pada *PHPMYAdmin* yang terdapat dalam *XAMPP (localhost)*.
4. *Web Service* dibangun menggunakan *PHP* yang akan menghasilkan format data berupa *JSON*.

1.5.4 Uji Coba dan Evaluasi

Tahap ini dapat dilakukan dengan beberapa uji coba, antara lain :

1. Sistem mampu menerima *request* dari *client* (lebih dari satu *client*) dan menampilkan data *last updateclient* beserta data lokasi dari masing-masing *client*
2. *Web* mampu menampilkan data lokasi dari masing-masing *client*

1.5.5 Penyusunan Laporan Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat.

1.6 Sistematika Penulisan

Sistematika penulisan buku tugas akhir secara garisbesar antara lain:

1. Bab 1. Pendahuluan
Bab ini meliputi latar belakang masalah, tujuan pembuatan Tugas Akhir, rumusan masalah, batasan

masalah, metodologi yang digunakan, dan sistematika penyusunan.

2. Bab 2. Tinjauan Pustaka

Bab ini meliputi dasar teori dan penunjang yang berkaitan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.

3. Bab 3. Perancangan Perangkat Lunak

Bab ini membahas perancangan dari sistem yang akan dibuat meliputi arsitektur dan proses perangkat lunak.

4. Bab 4. Implementasi Perangkat Lunak

Bab ini membahas implementasi dari desain sistem yang akan dilakukan pada tahap perancangan.

5. Bab 5. Uji Coba dan Evaluasi

Bab ini membahas uji coba dan evaluasi untuk mengetahui kemampuan perangkat lunak.

6. Bab 6. Penutup

Bab ini berisi kesimpulan dari hasil uji coba yang dilakukan serta saran untuk pengembangan lebih lanjut mengenai perangkat lunak.

BAB II

TINJAUAN PUSTAKA

Bab ini berisi penjelasan teori-teori yang berkaitan dengan pengimplementasian perangkat lunak. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap sistem yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

2.1 Android

Android adalah sistem operasi *mobile* yang berbasis *linux*, dimana *android* itu sendiri merupakan gabungan dari beberapa bahasa, yaitu *Java* sebagai bahasa *default*-nya dan *XML* sebagai bahasa *scripting*-nya.

Pada sistem pencatat data lokasi ini, *user*-lah yang mempunyai peranan paling penting dalam penggunaan aplikasi *android* ini. *User* adalah satu-satunya pengguna yang mempunyai hak akses terhadap aplikasi *android*. Berikut ini adalah pustaka pendukung yang digunakan dalam pembuatan aplikasi *android*-nya.

2.1.1 JSONParser

JSON Parser merupakan suatu *class* yang mempunyai peranan merubah tipe data *String* menjadi *JSON Object* maupun sebaliknya. Pada sistem pencatat data lokasi ini, *JSON Parser* digunakan ketika proses login, dimana tipe data *String* akan dirubah menjadi *JSON Object* dan akan di-*response* oleh *webservice* dan ketika proses *report last update* lokasi pada aplikasi *android*-nya. Keterangan lebih lanjut mengenai *JSON Parser* ini dapat dilihat pada *class JSONParser*.

2.1.2 Shared Preference

Shared Preference merupakan suatu *class* yang mempunyai peranan sebagai penyimpanan data primitif pada *file* internal aplikasi dengan pasangan variabel *key-value* [2]. Pada sistem pencatat data lokasi ini, *shared preference* digunakan

untuk menyimpan nilai *username* yang telah *login* melalui aplikasi *android*-nya. Tujuannya adalah agar sistem mengetahui siapa saja *user* yang sedang berinteraksi dengan sistem, melalui informasi *username*-nya tersebut. Keterangan lebih lanjut mengenai *shared preference* ini dapat dilihat pada *class SessionManager*.

2.1.3 AsyncTask

Asynctask merupakan suatu *class abstrak* yang berperan sebagai pengatur jalannya eksekusi suatu operasi pada *background* yang kemudian nantinya akan ditampilkan pada *user interface* (UI). *Asynctask* mempunyai 4 *method* utama, yaitu *onPreExecute()*, *doInBackground()*, *onProgressUpdate()*, dan *onPostExecute()*. Pada sistem pencatat data lokasi ini, *asynctask* digunakan ketika menjalankan proses *login* dan proses kirim kode lokasi. Keterangan lebih lanjut mengenai *asynctask* ini dapat dilihat pada *class LoginActivity*.

2.2 PHP

PHP (*HyperText Preprocessor*) merupakan bahasa *scripting* yang bersifat *server-side*, dimana nantinya perintah-perintah *PHP* akan dieksekusi di *server* dan ditampilkan dalam format *HTML* (*HyperText Markup Language*). Dengan karakteristik *PHP* yang demikian, maka kode program yang ditulis tidak akan terlihat oleh pengguna sehingga keamanan halaman web lebih terjamin.

PHP dapat berjalan pada berbagai macam *web server*, seperti *IIS* (*Internet Information Server*), *PWS* (*Personal Web Server*), *Apache*, dll. Selain itu, *PHP* juga mampu berjalan pada lintas *platform*. Artinya adalah *PHP* mampu berjalan pada berbagai macam sistem operasi, seperti *Windows*, *Linux*, *Macintosh*, dan lain-lain.

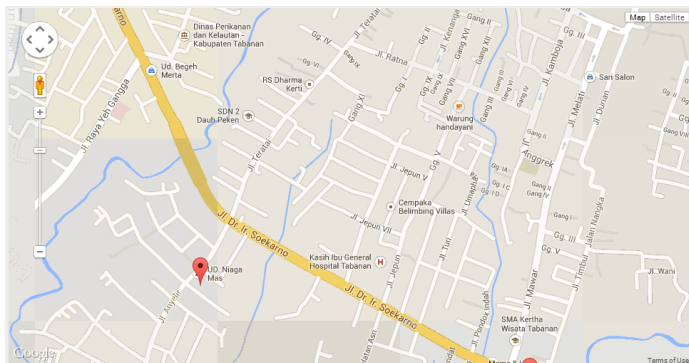
Salah satu keunggulan yang dimiliki oleh *PHP* adalah kemampuannya dalam melakukan koneksi ke berbagai macam jenis *database*, seperti *Oracle*, *Microsoft SQL Server*, *PostgreSQL*, dan lain-lain, sehingga dapat membuat suatu

halaman yang dinamis. *PHP* juga mempunyai kemampuan berkomunikasi dengan layanan (*service*) ataupun perangkat lain melalui protokol. Salah satu protokol yang paling umum digunakan adalah *HTTP* (*HyperText Transfer Protocol*). *HTTP* inilah protokol yang digunakan untuk menghubungkan *PHP* dengan *database* dan menghubungkan *android* dengan *database*.

2.3 JavaScript (Google Maps API)

Javascript merupakan bahasa *scripting* yang bersifat client-side, dimana nantinya perintah-perintah *javascript* dapat langsung dieksekusi tanpa harus terhubung dengan *server* terlebih dahulu. Dengan karakteristik *javascript* yang demikian, maka fungsifungsinya dapat disisipkan ke dalam *HTML* (*HyperText Markup Language*) secara langsung maupun diletakkan di suatu file teks dan di-link dari dokumen *HTML*-nya.

Google Maps API merupakan *library javascript* yang digunakan untuk menampilkan peta digital pada halaman *HTML* [3]. Pada *google maps* juga akan ditampilkan *marker*, dimana *marker* itulah representasi dari lokasi yang harus dikunjungi *user*. *Marker* tersebut dikelompokkan berdasarkan *user*-nya, karena setiap *user* memiliki lokasi yang berbeda-beda. Tampilan sederhana *google maps API* dapat dilihat pada Gambar 2.1 berikut.



Gambar 2.1 Halaman Google Maps API

2.4 Web Service

Web service adalah aplikasi, perangkat lunak, atau bagian dari perangkat lunak yang dapat diakses oleh berbagai macam perangkat, melalui suatu protokol tertentu (dalam kasus ini protokol *HTTP*). *Web service* ini sendiri juga merupakan suatu metode pertukaran data, tanpa memperhatikan dimana database ditanamkan, dibuat dalam bahasa pemrograman apakah aplikasi tersebut, serta pada platform apa data tersebut dikonsumsi atau diolah.

Pada sistem pencatatan data lokasi ini, menggunakan *web service* dilakukan pada saat menghubungkan *database MySQL* dengan aplikasi *android*-nya. Data yang dikirimkan dari aplikasi *android* akan di-*response* oleh *web service* sebelum akhirnya diterima oleh *database* [1]. Data yang di-*response* tersebut akan dikembalikan dengan format *JSON (Javascript Notation Object)*. Itulah mengapa pada penjelasan sebelumnya terdapat *JSON Parser* pada aplikasi *android*-nya.

BAB III

PERANCANGAN PERANGKAT LUNAK

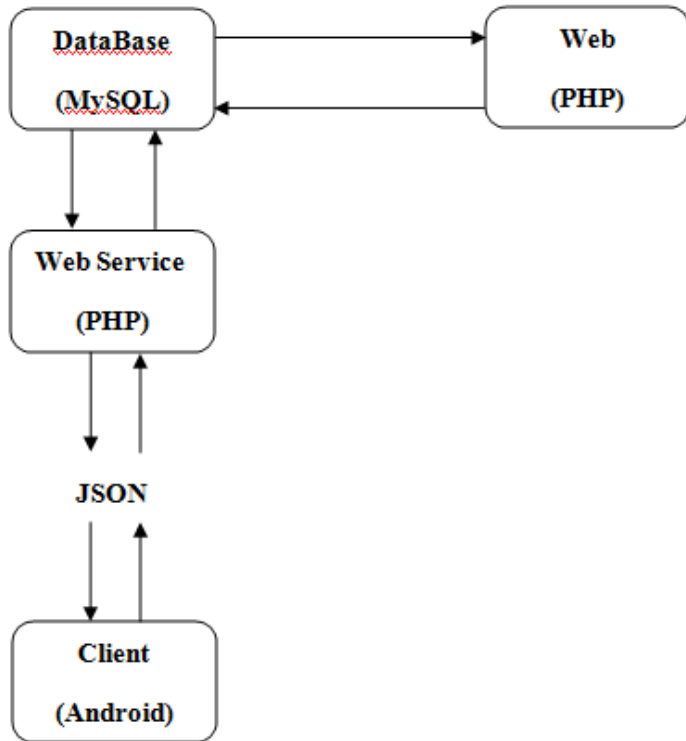
Perancangan perangkat lunak merupakan tahap yang menjelaskan dan menggambarkan bagaimana perangkat lunak bekerja sesuai dengan sistem yang hendak dibangun. Pada bab ini akan dibahas mengenai analisis dan perancangan dari penerapan sistem pencatatan data lokasi ini.

3.1 Deskripsi Umum

Padasub bab ini akan dibahas mengenai tahap analisis kebutuhan serta desain dari sistem yang akan dibangun. Tahap ini membahas mengenai analisis kebutuhan yang menjadi dasar pada tahap perancangan dan tentunya akan berlanjut dengan implementasi, yang tentunya akan dibahas pada bab berikutnya. Analisis kebutuhan dan desain akan disesuaikan dengan materi perkuliahan yang telah didapat, sehingga dapat digunakan serta dipahami oleh dosen, mahasiswa, maupun pihak lain yang hendak menggunakan aplikasi ini.

3.2 Arsitektur Aplikasi

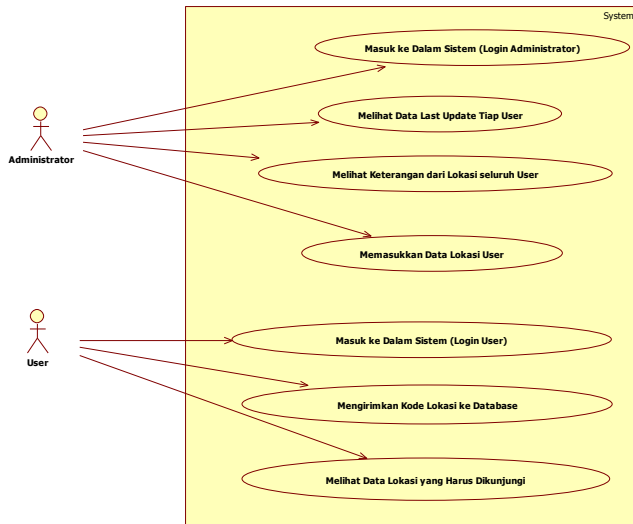
Sistem yang hendak dibangun ini menggunakan arsitektur *client to web*, dimana ada *web* yang ditangani oleh *administrator* dan *android* yang ditangani oleh *user*. Arsitektur aplikasi ini dapat dilihat pada Gambar 3.2 berikut.



Gambar 3.2 Arsitektur Aplikasi

3.3 Skenario Use Case

Pada diagram *use case* ini akan digambarkan proses bisnis yang berjalan pada sistem secara keseluruhan. Skema diagram *use case*-nya ditunjukkan pada Gambar 3.3 berikut.



Gambar 3.3 Use Case Diagram

Penjelasan mengenai diagram use case ini akan dijabarkan lagi pada Tabel 3.1 berikut ini.

Tabel 3.1 Tabel Use Case

No	Kode Use Case	Nama Use Case	Keterangan
1	UC-001	Masuk ke dalam sistem (<i>Login Web</i>)	<i>Administrator</i> masuk ke dalam sistem (dalam kasus ini melalui <i>web</i>)
2	UC-002	Melihat data <i>last update</i> setiap user	<i>Administator</i> dapat memantau apakah user telah mengunjungi lokasi berdasarkan <i>last update</i> -nya

No	Kode Use Case	Nama Use Case	Keterangan
3	UC-003	Melihat keterangan dari lokasi yang dikunjungi user	<i>Administrator</i> dapat mengetahui lokasi yang dikunjungi user berdasarkan peta yang ditampilkan (<i>google maps</i>)
4	UC-004	Memasukkan data lokasi user	<i>Administrator</i> dapat memasukkan data lokasi yang baru apabila nanti terdapat penambahan lokasi
5	UC-005	Masuk ke dalam sistem (<i>Login Android</i>)	<i>User</i> masuk ke dalam sistem (dalam kasus ini melalui <i>android</i>)
6	UC-006	Mengirimkan data lokasi ke <i>database</i> (via <i>android</i>)	<i>User</i> mengirimkan data lokasinya (kode lokasi pelanggan) ke <i>database</i>
7	UC-007	Melihat data lokasi yang harus dikunjungi (via <i>web</i>)	<i>User</i> melihat lokasi yang harus dikunjungi via <i>web</i>

3.4 Fitur Aplikasi

Aplikasi yang hendak dikembangkan dalam Tugas Akhir ini merupakan salah satu bentuk implementasi dari sistem pencatat data lokasi sederhana, dimana pada dasarnya sistem ini mengadaptasi penerapan *web* dan *android* yang diintegrasikan dalam *network*. Fitur-fitur yang tersedia dalam aplikasi ini diharapkan bisa membantu pengguna, utamanya *administrator* dalam monitoring dan penyimpanan data, sehingga dapat mendukung terciptanya *transparansi* sistem.

Fitur-fitur yang terdapat dalam *web* antara lain sebagai berikut :

1. Menu *login web*
2. Menu utama *administrator*
3. Menu *entryadministrator*
4. Menu peta *administrator*
5. Menu peta *user*

Fitur-fitur yang terdapat dalam *android* antara lain sebagai berikut :

1. Menu *login android*
2. Menu *entry android*

3.5 Proses Utama

Proses-proses yang terdapat dalam sistem ini menggambarkan bagaimana fungsionalitas dari sistem telah berjalan sebagaimana mestinya. Berikut ini adalah bagaimana proses-proses utamanya berlangsung.

3.5.1 Proses Pada Web

Berikut ini akan dijabarkan mengenai proses-proses yang dilakukan oleh *administrator* maupun *user* pada *web*.

3.5.1.1 Masuk Ke Dalam Sistem (LoginWeb)

Proses yang dilakukan adalah *administrator* melakukan login terlebih dahulu untuk bisa masuk dan memanfaatkan fitur yang ada di dalam sistem.

3.5.1.2 Melihat Data Last Update Setiap User

Proses yang dilakukan adalah *administrator* melihat data *last update* dari *user* berdasarkan data yang telah disimpan dan di-*query* oleh *database*.

3.5.1.3 Melihat Keterangan Lokasi Seluruh User

Proses yang dilakukan adalah *administrator* melihat lokasi user berdasarkan koordinat lokasinya dan ditampilkan ke dalam peta digital (*google maps*).

3.5.1.4 Memasukkan Data Lokasi User

Proses yang dilakukan adalah *administrator* menambahkan data lokasi baru apabila terjadi penambahan lokasi *user*.

3.5.1.5 Melihat Data Lokasi yang Harus Dikunjungi

Proses yang dilakukan adalah *user* bisa melihat daftar lokasi yang harus dikunjungi. Lokasi *user* akan ditampilkan dengan peta digital (*google maps*).

3.5.2 Proses Pada Android

Berikut ini akan dijabarkan mengenai proses-proses yang dilakukan oleh *user* pada *mobile android*.

3.5.2.1 Masuk ke Dalam Sistem (Login Android)

Proses yang dilakukan adalah *user* melakukan login terlebih dahulu untuk bisa masuk dan memanfaatkan fitur yang ada di dalam sistem.

3.5.2.2 Mengirimkan Data Lokasi Ke Database

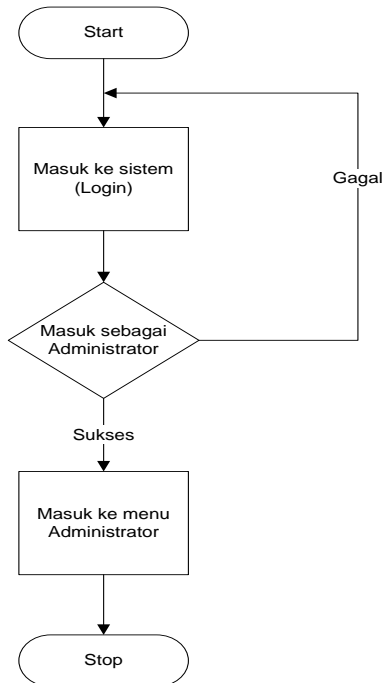
Proses yang dilakukan adalah *user* mengirimkan datalokasinya berdasarkan lokasi dimana dia berada saat ini.

3.6 Diagram Alir

3.6.1 Diagram Alir Masuk Ke Dalam Sistem (Login Web)

Pada proses ini, *administrator* mempunyai hak akses khusus untuk masuk ke dalam sistem (*web*). Setelah masuk ke sistem, *administrator* mampu mengakses fitur-fitur yang khusus dimiliki oleh *administrator*, yaitu menu utama *administrator*,

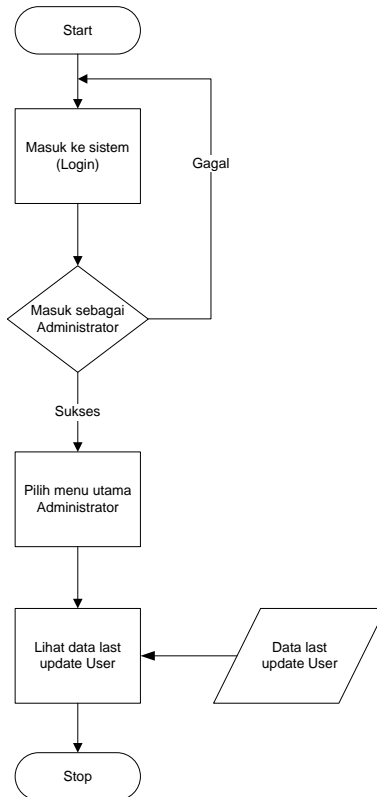
menu entry administrator dan *menu peta administrator*. Diagram alir proses ini dapat dilihat pada Gambar 3.4.



Gambar 3.4 Diagram Alir Masuk Ke Dalam Sistem

3.6.2 Diagram Alir Melihat Data Last Update Setiap User

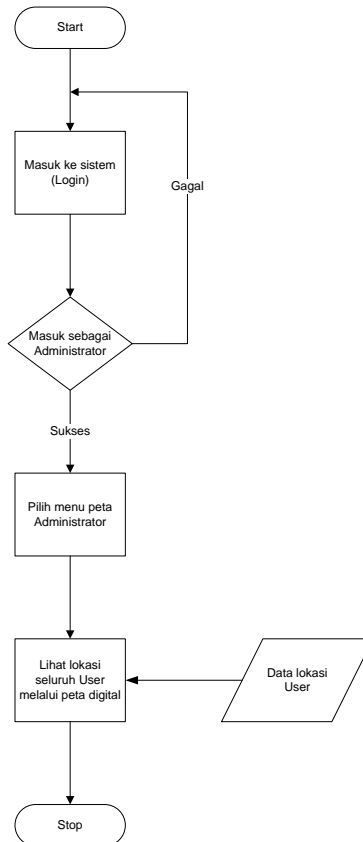
Pada proses ini, *administrator* akan mengakses menu utama *administrator*, tentunya setelah *administrator* berhasil masuk ke sistem. Diagram alir proses ini dapat dilihat pada Gambar 3.5.



Gambar 3.5 Diagram Alir Melihat Data Last Update Setiap User

3.6.3 Diagram Alir Melihat Keterangan Lokasi Seluruh User

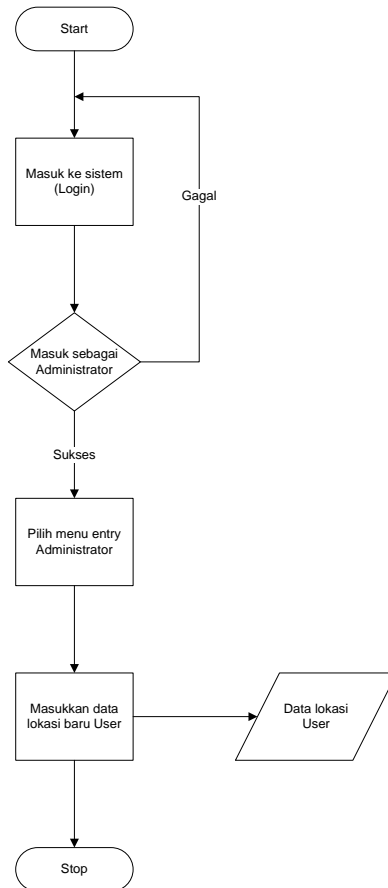
Pada proses ini, *administrator* akan mengakses fitur menu peta *administrator*, dimana data lokasi dari seluruh user akan ditampilkan pada halaman *maps*. Diagram alir proses ini dapat dilihat pada Gambar 3.6.



Gambar 3.6 Diagram Alir Melihat Keterangan Lokasi Seluruh User

3.6.4 Diagram Alir Memasukkan Data Lokasi User

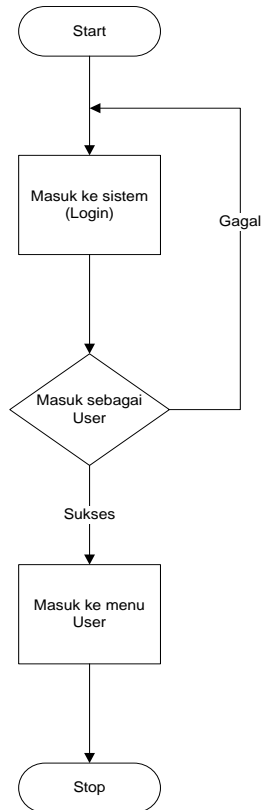
Pada proses ini, *administrator* akan mengakses menu *entryadministrator*, tentunya setelah *administrator* berhasil masuk ke sistem. Diagram alir proses ini dapat dilihat pada Gambar 3.7.



Gambar 3.7 Diagram Alir Memasukkan Data Lokasi User

3.6.5 Diagram Alir Masuk Ke Dalam Sistem (Login Android)

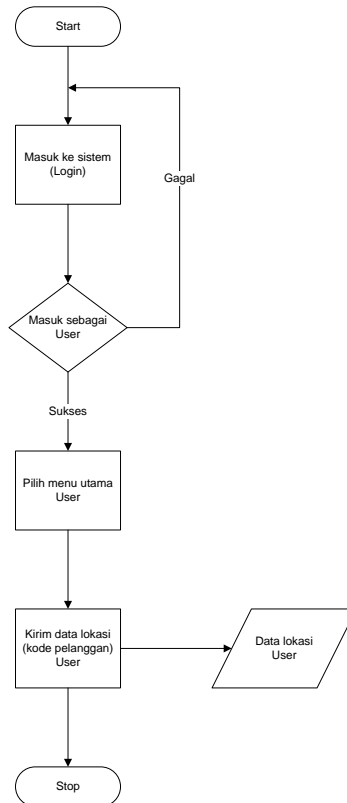
Pada proses ini, *user* mempunyai hak akses khusus untuk masuk ke dalam sistem (*android*). Setelah masuk ke sistem, *user* mampu mengakses fitur yang dimiliki oleh *user*, yaitu menu utama *user*. Diagram alir proses ini dapat dilihat pada Gambar 3.8.



Gambar 3.8 Diagram Alir Masuk ke Dalam Sistem (Login Android)

3.6.6 Diagram Alir Mengirimkan Data Lokasi Ke Database

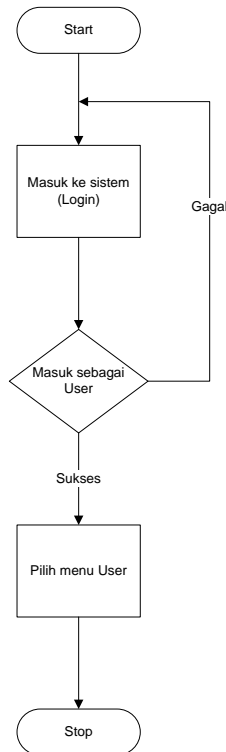
Pada proses ini, *user* akan mengakses menu utama user melalui *device android*. Setiap *user* mempunyai hak akses untuk masuk ke dalam sistem. Pada menu utama *user* inilah data lokasi akan di-*entry* berdasarkan lokasi pelanggan *user*. Diagram alir proses ini dapat dilihat pada Gambar 3.9.



Gambar 3.9 Diagram Alir Mengirimkan Data Lokasi ke Database

3.6.7 Diagram Alir Melihat Keterangan Lokasi Setiap User

Pada proses ini, kondisi *user* akan mengakses menu peta *user* melalui *web*. Setiap *user* bisa melihat lokasi mana saja yang harus dikunjungi melalui halaman *maps*. Diagram alir proses ini dapat dilihat pada Gambar 3.10.



Gambar 3.10 Diagram Alir Melihat Keterangan Lokasi Setiap User

3.7 Rancangan Antarmuka

Aplikasi sistem pencatat data lokasi ini merupakan integrasi antara aplikasi *web* dan aplikasi *android*. Untuk menjalankan aplikasi ini, maka diperlukan gambaran yang jelas mengenai fitur-fitur apa saja yang diadaptasi ke dalam sistem. Tentunya, dibutuhkan semacam antarmuka yang bisa menjelaskan bagaimana sistem berjalan. Rancangan antarmuka inilah yang

akhirnya dibuat untuk memudahkan pengguna dalam menjalankan fitur-fitur yang ada, dalam hal ini *administrator* dan *user*.

3.7.1 Antarmuka Web

3.7.1.1 Antarmuka Halaman Beranda Login Web

Pengguna yang bisa mengakses halaman ini adalah semua pengguna (dalam hal ini *administrator* dan *user*). Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.11.



The image shows a web page titled "Halaman Login". Inside the page, there is a box containing the text "Silakan Login dahulu". Below this text are two input fields: "Email" and "Password". At the bottom of the box are two buttons: "Login" and "Register".

Gambar 3.11 Antarmuka Halaman Beranda Login

3.7.1.2 Antarmuka Halaman Menu Utama Administrator

Pengguna yang bisa mengakses halaman ini adalah *administrator*. Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.12.

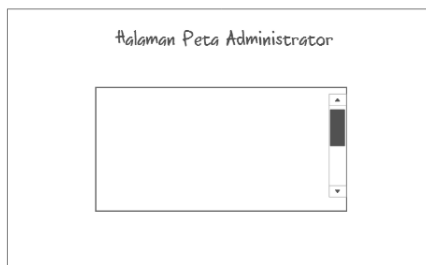
Halaman Last Update User

Nama User	Kode Lokasi	Last Update

Gambar 3.12 Antarmuka Halaman Menu Utama Administrator

3.7.1.3 Antarmuka Halaman Menu Peta Administrator

Pengguna yang bisa mengakses halaman ini adalah *administrator*. Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.13



Gambar 3.13 Antarmuka Halaman Menu Peta Administrator

3.7.1.4 Antarmuka Halaman Menu Entry Administrator

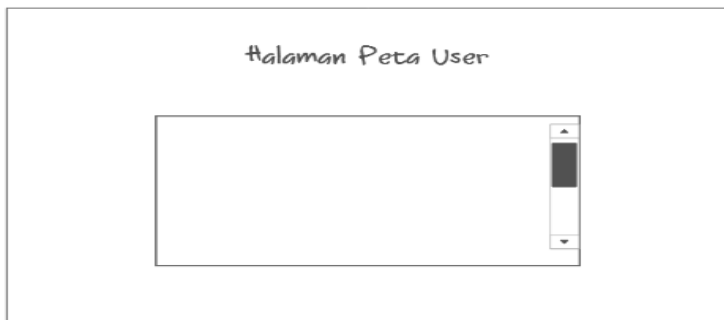
Pengguna yang bisa mengakses halaman ini adalah *administrator*. Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.14



Gambar 3.14 Antarmuka Halaman Menu Entry Administrator

3.7.1.5 Antarmuka Halaman Menu Peta User

Pengguna yang bisa mengakses halaman ini adalah *user*. Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.15.



Gambar 3.15 Antarmuka Halaman Menu Peta User

3.7.2 Antarmuka Android

3.7.2.1 Antarmuka Halaman Beranda Login Android

Pengguna yang bisa mengakses halaman ini adalah *user*. Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.16.



The screenshot shows a light blue rectangular frame representing the login page. At the top center, the text "Halaman Login" is displayed. Below this, there are two input fields: the first is labeled "Username" and the second is labeled "Pasword". Both labels are positioned to the left of the input lines. At the bottom center of the frame, there is a white rectangular button with the text "Login" in black.

Gambar 3.16 Antarmuka Halaman Beranda Login

3.7.2.2 Antarmuka Halaman Menu Utama User

Pengguna yang bisa mengakses halaman ini adalah *user*. Rancangan antarmuka halaman ini bisa dilihat pada Gambar 3.17.



The screenshot shows a light blue rectangular frame representing the main menu page. At the top center, the text "Menu Utama User" is displayed. Below this, there are two input fields: the first is labeled "Angka Meter" and the second is labeled "Kode Lokasi". Both labels are positioned to the left of the input lines. At the bottom center of the frame, there is a white rectangular button with the text "Kirim" in black.

Gambar 3.17 Antarmuka Halaman Menu Utama User

BAB IV

IMPLEMENTASI PERANGKAT LUNAK

Bab ini membahas implementasi perancangan perangkat lunak yang merupakan penerapan sistem pencatatan data lokasi yang mengacu pada perancangan yang telah dibahas sebelumnya. Selain itu, bab ini juga membahas lingkungan pembangunan perangkat lunak yang menjelaskan spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam pembangunan sistem.

4.1 Lingkungan Implementasi

Dalam implementasinya, sistem pencatat data lokasi ini didukung dengan lingkungan perangkat keras dan perangkat lunak sebagai berikut.

4.1.1 Lingkungan Perangkat Lunak

Adapun beberapa perangkat lunak yang digunakan selama proses implementasi, yaitu:

- *Android* sebagai sistem operasi pada *mobile* yang digunakan untuk membangun aplikasi pada *user*.
- *Microsoft Windows 7 32-bit* sebagai sistem operasi pada *web* yang digunakan untuk membangun aplikasi pada *administrator*.
- *XAMPP*, dimana *Apache* sebagai *WebServer (localhost)* dan *MySQL (PhpMyAdmin)* sebagai *database*-nya.
- *Google Maps API* sebagai pustaka *Javascript* yang digunakan untuk menampilkan peta digital *user* maupun *administrator*.
- *Web Service* sebagai penghubung aplikasi *Android* dengan *database MySQL*.

4.1.2 Lingkungan Perangkat Keras

Adapun perangkat keras yang digunakan selama proses implementasi, yaitu :

- Laptop, Processor Intel Core2Duo, RAM 2 GB, Sistem Operasi *Windows 7* 32-bit.
- *Android* Smartphone, Sony Xperia Go RAM 512 MB, Sistem Operasi *Android Jelly Bean 4.2.1*

4.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak pada sistem pencatat data lokasi ini dilakukan pada 2 aplikasi, yaitu aplikasi *web* dan aplikasi *android*. Kedua aplikasi ini terhubung dengan *database* yang sama, yaitu *MySQL*. Aplikasi *web* terhubung ke *MySQL* secara langsung, sedangkan aplikasi *android* terhubung ke *MySQL* dengan *web service PHP*.

4.2.1 Implementasi Web

Implementasi pada *web* ini melibatkan dua pengguna, yaitu *administrator* dan *user*. Penjelasan lebih lanjut mengenai implementasinya dapat dilihat pada penjelasan di bawah ini.

4.2.1.1 Implementasi Login Web

Implementasi *Login Web* bertujuan untuk mendata siapa saja pengguna yang masuk ke dalam sistem. Sistem hanya menerima hak akses untuk 2 macam pengguna saja, yaitu *administrator* dan *user*. Tampilan halaman *login web* dapat dilihat pada Gambar 4.18 berikut.



Gambar 4.18 Halaman Login Web

Pseudocode Login Web ini dapat dilihat pada Gambar 4.19 berikut.

	<div><div>Halaman Login Web</div><div>start</div><div>Deklarasi</div><div>username : String</div><div>password : String</div><div>jabatan : String</div><div>Deskripsi</div><div>Read (username, password, jabatan)</div><div>If(jabatan == admin)</div><div> Redirect to 'halaman admin'</div><div>If(jabatan == user)</div><div> Redirect to 'halaman user'</div><div>end</div></div>
--	---

Gambar 4.19 Pseudocode Halaman Login Web

4.2.1.2 Implementasi Menu Utama Administrator

Implementasi *Menu Utama Administrator* berlaku untuk *administrator* yang telah masuk ke dalam sistem. Pada menu ini akan ditampilkan data *last update* per user. Tampilan halaman

menu utama administrator dapat dilihat pada Gambar 4.20 berikut.

Last Update	Nama Pelanggan	Angka Meter	Nama User
2014-06-09 07:17:16	F	100	susi
2014-07-21 00:19:40	A	0	test

Lihat Statistik >>>

Gambar 4.20 Halaman Menu Utama Administrator

Pseudocode Menu Utama Administrator ini dapat dilihat pada Gambar 4.21 berikut.

	<pre>Halaman Menu Utama Administrator start Deklarasi If(jabatan == admin) Select menu utama administrator Deskripsi Read (tabel last update user) end</pre>
--	--

Gambar 4.21 Pseudocode Halaman Menu Utama Administrator

4.2.1.3 Implementasi Menu Peta Administrator

Implementasi Menu Peta Administrator berlaku untuk administrator yang telah masuk ke dalam sistem. Pada menu ini akan ditampilkan peta digital (google maps) yang berisikan informasi lokasi dari seluruh user. Informasi lokasi ini berupa marker pada google maps yang menunjukkan lokasi mana saja yang harus dikunjungi user. Menu ini bertujuan agar administrator mengetahui lokasi mana saja yang harus dikunjungi oleh setiap user. Tampilan halaman menu peta administrator dapat dilihat pada Gambar 4.22 berikut.



Gambar 4.22 Halaman Menu Peta Administrator

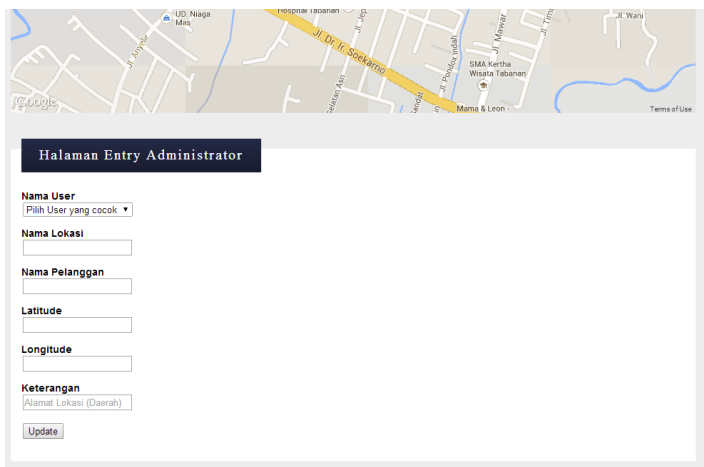
Pseudocode Menu Peta Administrator ini dapat dilihat pada Gambar 4.23 berikut.

	<pre> Halaman Menu Peta Administrator start Deklarasi If (jabatan == admin) Select menu peta administrator Deskripsi Read (halaman peta administrator) end </pre>
--	--

Gambar 4.23 Pseudocode Halaman Menu Peta Administrator

4.2.1.4 Implementasi Menu Entry Administrator

Implementasi *Menu Entry Administrator* berlaku untuk *administrator* yang telah masuk ke dalam sistem. Pada menu ini akan ditampilkan peta digital (*google maps*) beserta *form menu* untuk *men-entry* lokasi baru dari *user*. Menu ini bertujuan untuk memasukkan/menambah data lokasi seandainya ada penambahan lokasi yang harus dikunjungi oleh *user*. Tampilan halaman *menu entry administrator* dapat dilihat pada Gambar 4.24 berikut.



Gambar 4.24 Halaman Menu Entry Administrator

Pseudocode Menu Entry Administrator ini dapat dilihat pada Gambar 4.25 berikut.

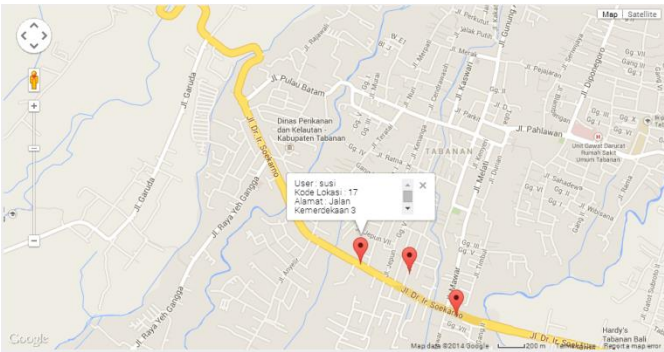
	<pre>Halaman Menu Entry Administrator start Deklarasi If (jabatan == admin) Select menu entry administrator Deskripsi Read (halaman entry administrator) end</pre>
--	--

Gambar 4.25 Pseudocode Halaman Menu Entry Administrator

4.2.1.5 Implementasi Menu Peta User

Implementasi Menu Peta *User* berlaku untuk *user* yang telah masuk ke dalam sistem. Pada menu ini akan ditampilkan peta digital (*google maps*) yang berisikan informasi lokasi dari *user* yang sebelumnya telah *login*. Informasi lokasi ini berupa *marker* pada *google maps* yang menunjukkan lokasi mana saja yang harus dikunjungi *user*. Menu ini bertujuan agar *user*

mengetahui lokasi mana saja yang harus dikunjungi. Tampilan halaman *menu peta user* dapat dilihat pada Gambar 4.26 berikut.



Gambar 4.26 Halaman Menu Peta User

Pseudocode Menu Peta User ini dapat dilihat pada Gambar 4.27 berikut.

	<div><div>Halaman Menu Peta</div><div>start</div><div>Deklarasi</div><div> If(jabatan == user)</div><div> Select menu peta user</div><div>Deskripsi</div><div> Read (halaman peta user)</div><div>end</div></div>
--	--

Gambar 4.27 Pseudocode Halaman Menu Peta User

4.2.2 Implementasi Android

Implementasi pada *android* ini hanya melibatkan satu pengguna, yaitu *user*. Penjelasan lebih lanjut mengenai implementasinya dapat dilihat pada penjelasan di bawah ini.

4.2.2.1 Implementasi Login Android

Implementasi *Login Android* bertujuan untuk mendata siapa saja *user* yang masuk ke dalam sistem. Sistem hanya menerima hak akses untuk *user* saja. Tampilan halaman *login android* dapat dilihat pada Gambar 4.28 berikut.



Gambar 4.28 Halaman Login Android


Pseudocode Menu Login User ini dapat dilihat pada Gambar 4.29 berikut.

	<pre>Halaman Login Android start Deklarasi username : String password : String jabatan : String Deskripsi Read (username, password, jabatan) If(jabatan == user) Redirect to 'halaman entry user'end</pre>
--	--

Gambar 4.29 Pseudocode Halaman Menu Login User

4.2.2.2 Implementasi Menu Entry User

Implementasi *Menu Entry User* berlaku untuk *user* yang telah masuk ke dalam sistem. Pada menu ini akan ditampilkan *menu form* yang akan digunakan *user* untuk meng-*entry* lokasi beserta angka meter air, dimana lokasi (dalam kasus ini lokasi merupakan kode pelanggan yang dikunjungi *user*) dan angka meter air tersebut merupakan bentuk pelaporan (*report*) yang hasilnya akan diterima di *database*. Sehingga, *administrator* juga bisa mengetahui data lokasi *user* (*last update*). Tampilan halaman *entry user* dapat dilihat pada Gambar 4.30 berikut.



The image shows a mobile application interface titled "Menu User". At the top, there is a status bar with the time "11:30 PM" and various icons. Below the title, a greeting "Halo, test" is displayed. The main form consists of two input fields. The first field is labeled "Masukkan angka meter" and contains a text input box with a vertical cursor. The second field is labeled "Masukkan Kode Pelanggan" and features a dropdown menu with the letter "A" selected. Below these fields is a "Submit" button.

Gambar 4.30 Halaman Menu Entry User

Pseudocode Menu Entry User ini dapat dilihat pada Gambar 4.31 berikut.

	Halaman Menu Entry User start Deklarasi If(jabatan == user) angka meter : String Kode lokasi pelanggan : String Deskripsi Read (jabatan, angka meter, jabatan, Kode lokasi pelanggan) Send_to_database End
--	---

Gambar 4.31 Pseudoode Halaman Menu Entry User

BAB V

UJI COBA DAN EVALUASI

Bab ini membahas uji coba dan evaluasi sistem pencatatan data lokasi ini. Sistem ini diuji dari segi fungsionalitas dan performanya. Pengujian akan dilakukan sesuai dengan perancangan perangkat lunak dan implementasi perangkat lunak yang semuanya telah dijelaskan pada bagian sebelumnya.

5.1 Lingkungan Uji Coba

Uji coba ini dilakukan dengan menggunakan *mobile android* yang telah terkoneksi dengan *network* dan satu buah laptop yang telah terinstall *webserver (localhost)*. Berikut ini adalah spesifikasi perangkat yang digunakan :

- Laptop
 - Spesifikasi Perangkat Keras
 - *Processor Intel Core2 Duo*
 - *RAM 2 GB*
 - *Sistem Operasi Windows 7 32-bit*
 - Spesifikasi Perangkat Lunak
 - *Apache Web Server*
 - *Database MySQL*
 - *Google Chrome Browser*
 - *Javascript Google Maps API version 3*
 - *Web Service PHP (format JSON)*
 - Spesifikasi Jaringan Lokal
 - *Alamat IP 192.168.1.113*
- *Mobile Android*
 - Spesifikasi Perangkat Keras
 - *Processor : Dual Core 1GHz Cortex-A9*
 - *RAM : 2 GB*
 - *Sistem Operasi Jelly Bean 4.2.1*

5.2 Skenario Uji Coba

Uji coba ini dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap perancangan benar-benar diimplementasikan dan bekerja seperti yang seharusnya. Uji coba akan didasarkan pada fitur-fiturnya untuk menguji kesesuaian respon sistem.

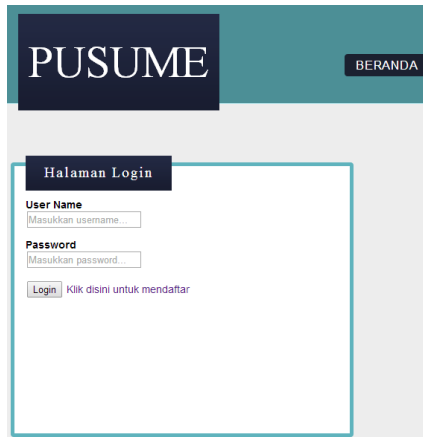
5.2.1 Uji Coba Fungsionalitas

Uji coba fungsionalitas dilakukan untuk melihat apakah proses utama yang berlangsung pada web maupun android berjalan sebagaimana mestinya. Uji coba fungsionalitas meliputi semua *use case* yang telah dijelaskan pada Bab 3 :

1. Uji coba masuk ke dalam sistem (*Login Web*).
2. Uji coba melihat data last update setiap *user*.
3. Uji coba melihat keterangan lokasi seluruh *user*.
4. Uji coba memasukkan data lokasi *user*.
5. Uji coba masuk ke dalam sistem (*Login Android*).
6. Uji coba mengirimkan kode lokasi ke *database*.
7. Uji coba melihat keterangan lokasi setiap *user*.

5.2.1.1 Uji Coba Masuk Ke Dalam Sistem (Login Web)

Uji coba ini dilakukan oleh *administrator* maupun *user* yang ingin masuk ke dalam sistem melalui halaman *web*, seperti yang ditunjukkan pada Gambar 5.32, Gambar 5.33, dan Gambar 5.34. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.2.



PUSUME **BERANDA**

Halaman Login

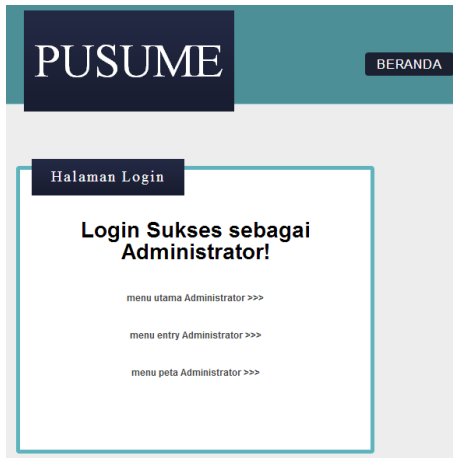
User Name
Masukkan username...

Password
Masukkan password...

Login [Klik disini untuk mendaftar](#)

Gambar 5.32 Uji Coba Masuk Ke Dalam Sistem (Login Web)

Tampilan halaman *web* ketika pengguna sukses masuk ke sistem sebagai *Administrator* dapat dilihat pada Gambar 5.33 berikut.



PUSUME **BERANDA**

Halaman Login

Login Sukses sebagai Administrator!

menu utama Administrator >>>

menu entry Administrator >>>

menu peta Administrator >>>

Gambar 5.33 Berhasil Masuk Ke Dalam Sistem (Administrator)

Tampilan halaman *web* ketika pengguna sukses masuk ke sistem sebagai *user* dapat dilihat pada Gambar 5.34 berikut.



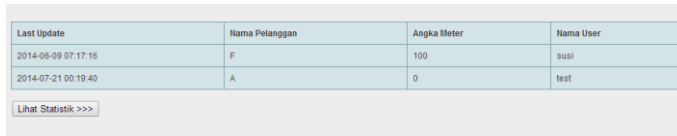
Gambar 5.34 Berhasil Masuk Ke Dalam Sistem (User)

Tabel 5.2 Uji Coba Masuk Ke Dalam Sistem

ID	UJ-01
Referensi Use Case	UC-001
Nama	Uji coba masuk ke dalam sistem (<i>Login Web</i>)
Tujuan Uji Coba	Menguji fitur menu <i>login web</i>
Kondisi awal	Sistem menampilkan <i>halaman login web</i> untuk diisi dengan <i>username</i> dan <i>password</i>
Skenario	Pengguna memilih menu “login”
Masukan	<i>username, password</i>
Keluaran	Pengguna berhasil masuk ke dalam sistem
Hasil Uji Coba	BERHASIL

5.2.1.2 Uji Coba Melihat Data Last Update Setiap User

Uji coba ini dilakukan oleh *administrator* yang telah berhasil masuk ke dalam sistem melalui halaman *web*, seperti yang ditunjukkan pada Gambar 5.35. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.3.



Last Update	Nama Pelanggan	Angka Meter	Nama User
2014-06-09 07:17:16	F	100	susi
2014-07-21 00:19:40	A	0	test

Lihat Statistik >>>

Gambar 5.35 Uji Coba Melihat Data Last Update Setiap User

Tabel 5.3 Uji Coba Melihat Data Last Update Setiap User

ID	UJ-02
Referensi Use Case	UC-002
Nama	Uji coba melihat data <i>last update</i> setiap <i>user</i>
Tujuan Uji Coba	Menguji fitur menu utama <i>administrator</i>
Kondisi awal	Sistem menampilkan tabel yang berisikan data <i>last update</i> dari masing-masing <i>user</i>
Skenario	<i>Administrator</i> memilih menu “menu utama <i>administrator</i> ”
Masukan	Tidak ada, hanya perlu <i>login</i> sebagai <i>administrator</i> terlebih dahulu
Keluaran	Data <i>last update</i> setiap user ditampilkan
Hasil Uji Coba	BERHASIL

5.2.1.3 Uji Coba Melihat Keterangan Lokasi Seluruh User

Uji coba ini dilakukan oleh *administrator* yang telah berhasil masuk ke dalam sistem melalui halaman *web*, seperti

yang ditunjukkan pada Gambar 5.36. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.4.



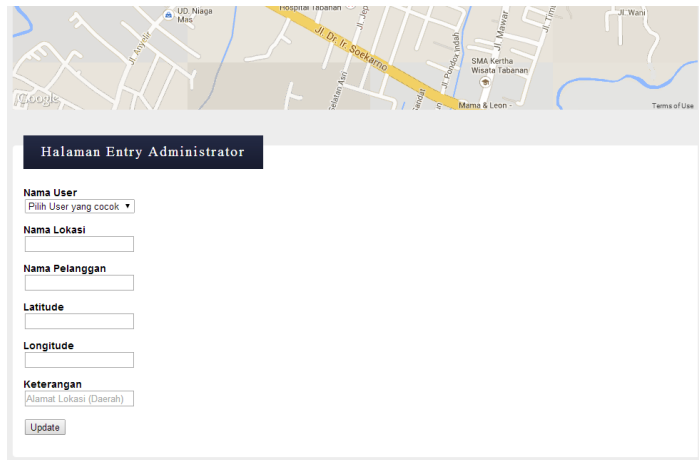
Gambar 5.36 Uji Coba Melihat Keterangan Lokasi Seluruh User

Tabel 5.4 Uji Coba Melihat Keterangan Lokasi Seluruh User

ID	UJ-03
Referensi Use Case	UC-003
Nama	Uji coba melihat keterangan lokasi seluruh <i>user</i>
Tujuan Uji Coba	Menguji fitur menu peta <i>administrator</i>
Kondisi awal	Sistem menampilkan halaman <i>google maps</i> yang berisikan data lokasi berupa <i>marker</i> yang merepresentasikan lokasi dari seluruh <i>user</i>
Skenario	<i>Administrator</i> memilih menu “menu peta <i>administrator</i> ”
Masukan	Tidak ada, hanya perlu <i>login</i> sebagai <i>administrator</i> terlebih dahulu
Keluaran	Data lokasi berupa peta digital <i>user</i> ditampilkan
Hasil Uji Coba	BERHASIL

5.2.1.4 Uji Coba Memasukkan Data Lokasi User

Uji coba ini dilakukan oleh *administrator* yang telah berhasil masuk ke dalam sistem melalui halaman *web*, seperti yang ditunjukkan pada Gambar 5.37. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.5.



Halaman Entry Administrator

Nama User
 Pilih User yang cocok ▼

Nama Lokasi

Nama Pelanggan

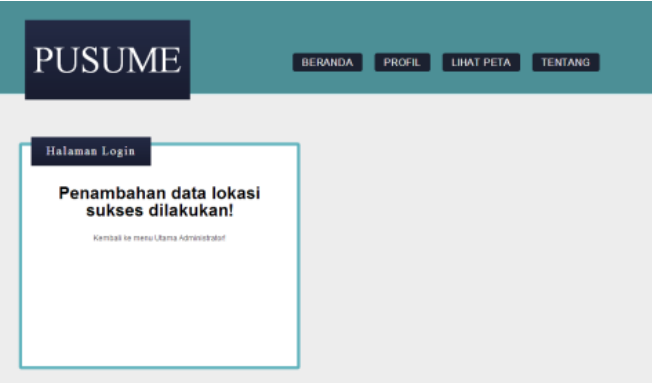
Latitude

Longitude

Keterangan
 Alamat Lokasi (Daerah)

Gambar 5.37 Uji Coba Memasukkan Data Lokasi User

Pada uji coba ini, user akan memasukkan data lokasi user. Tampilan halaman *web* ketika *Administrator* sukses menambah data lokasi *user* dapat dilihat pada Gambar 5.38 berikut.



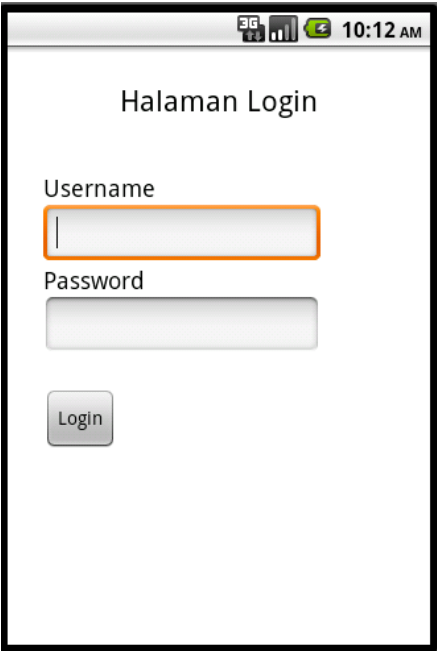
Gambar 5.38 Berhasil Memasukkan Data Lokasi User

Tabel 5.5 Uji Coba Memasukkan Data Lokasi User

ID	UJ-04
Referensi Use Case	UC-004
Nama	Uji coba memasukkan data lokasi <i>user</i>
Tujuan Uji Coba	Menguji fitur menu <i>entryadministrator</i>
Kondisi awal	Sistem menampilkan halaman <i>google maps</i> beserta <i>menu formentry</i> lokasi
Skenario	Administrator memilih menu “menu entry administrator”
Masukan	<i>Tidak ada</i> , hanya perlu <i>login</i> sebagai <i>administrator</i> terlebih dahulu
Keluaran	Data lokasi berhasil ditambahkan
Hasil Uji Coba	BERHASIL

5.2.1.5 Uji Coba Masuk Ke Dalam Sistem (Login Android)

Uji coba ini dilakukan oleh *user* yang telah berhasil masuk ke dalam sistem melalui halaman *android*, seperti yang ditunjukkan pada Gambar 5.39. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.6.



Gambar 5.39 Uji Coba Masuk Ke Dalam Sistem (Login Android)

Tabel 5.6 Uji Coba Masuk Ke Dalam Sistem (Login Android)

ID	UJ-05
Referensi Use Case	UC-005
Nama	Uji coba masuk ke dalam sistem (<i>Login Android</i>)
Tujuan Uji Coba	Menguji fitur menu <i>login android</i>
Kondisi awal	Sistem menampilkan <i>halaman login android</i> untuk diisi dengan <i>username</i> dan <i>password</i> oleh <i>user</i>

ID	UJ-05
Masukan	<i>username, password</i>
Keluaran	<i>User berhasil masuk ke dalam sistem</i>
Hasil Uji Coba	BERHASIL

5.2.1.6 Uji Coba Mengirimkan Data Lokasi Ke Database

Uji coba ini dilakukan oleh *user* yang telah berhasil masuk ke dalam sistem melalui halaman *android*, seperti yang ditunjukkan pada Gambar 5.40. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.7.

The screenshot shows an Android application window with a status bar at the top displaying '3G', signal strength, battery level, and the time '11:30 PM'. The main content area has a title 'Menu User' and a subtitle 'Halo, test'. There are two text input fields: the first is labeled 'Masukkan angka meter' and is highlighted with an orange border; the second is labeled 'Masukkan Kode Pelanggan'. Below the second input field is a button labeled 'Submit'.

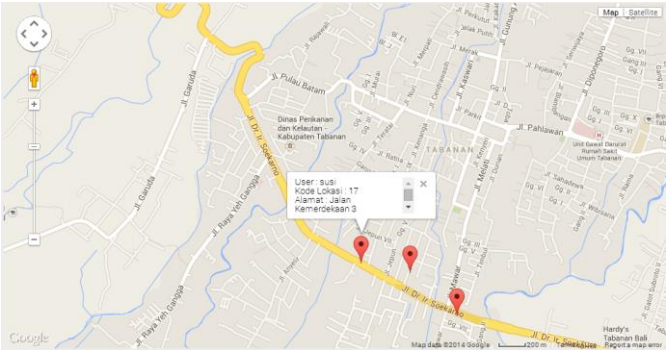
Gambar 5.40 Uji Coba Mengirimkan Data Lokasi Ke Database

Tabel 5.7 Uji Coba Mengirimkan Kode Lokasi Ke Database

ID	UJ-06
Referensi Use Case	UC-006
Nama	Uji coba mengirimkan kode lokasi ke <i>database</i>
Tujuan Uji Coba	Menguji fitur menu <i>entry android</i>
Kondisi awal	Sistem menampilkan <i>halaman entry user</i> untuk diisi dengan angka meter air dan kode lokasi
Skenario	User memilih menu “menu user” setelah berhasil masuk ke sistem
Masukan	angka meter, kode lokasi
Keluaran	angka meter dan kode lokasi berhasil terkirim ke <i>database</i>
Hasil Uji Coba	BERHASIL

5.2.1.7 Uji Coba Melihat Keterangan Lokasi Setiap User

Uji coba ini dilakukan oleh *user* yang telah berhasil masuk ke dalam sistem melalui halaman *web*, seperti yang ditunjukkan pada Gambar 5.41. Untuk penjelasan lebih lanjut mengenai uji cobanya, dapat dilihat pada Tabel 5.8.



Gambar 5.41 Uji Coba Melihat Keterangan Lokasi Setiap User

Tabel 5.8 Uji Coba Melihat Keterangan Lokasi Setiap User

ID	UJ-07
Referensi Use Case	UC-007
Nama	Uji coba melihat keterangan lokasi setiap <i>user</i>
Tujuan Uji Coba	Menguji fitur menu peta <i>user</i>
Kondisi awal	Sistem menampilkan halaman <i>google maps</i> yang berisikan data lokasi berupa <i>marker</i> yang merepresentasikan lokasi dari setiap <i>user</i>
Skenario	<i>User</i> memilih menu “menu peta <i>user</i> ”
Masukan	Tidak ada, hanya perlu <i>login</i> sebagai <i>user</i> terlebih dahulu
Keluaran	Data lokasi berupa peta digital <i>user</i> ditampilkan
Hasil Uji Coba	BERHASIL

5.2.2 Uji Coba Performa

Uji coba performa dilakukan untuk mengetahui seberapa lama waktu yang diperlukan sistem dalam menjalankan setiap proses utama yang telah dijelaskan pada Bab 3. Proses utama tersebut mewakili setiap fungsionalitas dari sistem ini.

5.2.2.1 Uji Coba Performa Pada Web

Pada tahap ini, ada lima hal yang dilakukan pada proses pengujian performa pada *web*. Pertama adalah pengujian terhadap proses masuk ke dalam sistem (*Login Web*), yang kedua adalah pengujian terhadap proses melihat data *last update* seluruh *user*, yang ketiga adalah pengujian terhadap proses melihat keterangan lokasi seluruh *user*, yang keempat adalah memasukkan data lokasi *user*, dan yang kelima adalah melihat keterangan lokasi setiap *user*. Untuk penjelasan lebih lanjut mengenai performa dari fungsionalitas aplikasidapat dilihat pada Tabel 5.9.

Tabel 5.9 Uji Coba Performa Pada Web

No	Fungsionalitas	Performa Yang Di Uji	<i>Time / Latency</i>
1	Masuk ke dalam sistem (<i>Login Web</i>)	Berhasil atau tidaknya pengguna (dalam kasus ini <i>administrator</i> maupun <i>user</i>) masuk ke dalam sistem	83 ms / 81 ms
2	Melihat data <i>last update</i> seluruh <i>user</i>	Menampilkan keseluruhan informasi data <i>last update</i> seluruh <i>user</i>	64 ms / 62 ms

No	Fungsionalitas	Performa Yang Di Uji	Time / Latency
3	Melihat keterangan lokasi seluruh <i>user</i>	Menampilkan keseluruhan informasi data lokasi <i>last update</i> seluruh <i>user</i> pada halaman peta digital (<i>google maps</i>)	23 ms / 21 ms
4	Memasukkan data lokasi <i>user</i>	Berhasil atau tidaknya <i>user</i> (dalam kasus ini <i>administrator</i>) memasukkan data lokasi <i>user</i> yang baru (<i>entry</i> data lokasi baru)	126 ms/ 124 ms
5	Melihat keterangan lokasi setiap <i>user</i>	Menampilkan data lokasi setiap <i>user</i> pada halaman peta digital (<i>google maps</i>)	49 ms/ 49ms

5.2.2.2 Uji Coba Performa Pada Android

Pada tahap ini, ada dua hal yang dilakukan pada proses pengujian performa pada *android*. Pertama adalah pengujian terhadap proses masuk ke dalam sistem (*Login Android*) dan yang kedua adalah mengirimkan data lokasi pelanggan ke *database*. Untuk penjelasan lebih lanjut mengenai performa dari fungsionalitas aplikasi dapat dilihat pada Tabel 5.10.

Tabel 5.10 Uji Coba Performa Pada Android

No	Fungsionalitas	Performa Yang Di Uji	Time / Latency
1	Masuk ke dalam sistem (<i>Login Android</i>)	Berhasil atau tidaknya pengguna (dalam kasus ini <i>user</i>) masuk ke dalam sistem	495 ms / 495 ms
2	Mengirimkan data lokasi pelanggan ke <i>database</i>	Berhasil atau tidaknya <i>user</i> mengirimkan data lokasi pelanggan ke <i>database</i>	185 ms / 185 ms

5.2.3 Uji Coba Kompatibilitas

Uji coba kompatibilitas dilakukan untuk mengetahui kesesuaian sistem *web* ketika digunakan pada berbagai *web browser* yang berbeda-beda oleh *administrator*, seperti yang ditunjukkan pada Tabel 5.11 tersebut.

Tabel 5.11 Uji Coba Kompatibilitas

Web Browser	Versi	Sistem Operasi	Keterangan
Google Chrome	24	Windows 7	Berhasil
Opera Mini	18	Windows 7	Berhasil
Mozilla Firefox	10	Windows 7	Berhasil
Internet Explorer	9	Windows 7	Berhasil

5.3 Evaluasi Uji Coba

Dari uji coba fungsional yang telah dilakukan terhadap aplikasi ini, maka dapat dibuat kesimpulan keberhasilan uji coba yang ditunjukkan oleh Tabel 5.12. Tabel tersebut menjelaskan ID masing-masing *use case* beserta ID uji coba yang telah dilakukan serta hasil yang didapatkan selama pengujian.

Tabel 5.12 Evaluasi Uji Coba Fungsionalitas

No	ID Use Case	ID Uji Coba	Hasil	Catatan
1	UC-01	UJ-01	Berhasil	-
2	UC-02	UJ-02	Berhasil	-
3	UC-03	UJ-03	Berhasil	-
4	UC-04	UJ-04	Berhasil	-
5	UC-05	UJ-05	Berhasil	-
6	UC-06	UJ-06	Berhasil	-
7	UC-07	UJ-07	Berhasil	-

Tabel 5.12 di atas menunjukkan bahwa antara desain perancangan sistem, implementasi, serta uji coba yang telah dilakukan telah sesuai dan dapat berjalan dengan baik yang dibuktikan dengan keberhasilan semua uji coba yang telah dilakukan.

Sistem telah melewati uji coba performa, dimana performanya diukur berdasarkan lama waktu dari fungsionalitas aplikasi, yaitu *web* dan *android*. Selain itu, sistem juga telah melewati uji coba kompatibilitas yaitu uji coba pada aplikasi *web browser* yang berbeda-beda, namun tetap bisa diakses oleh pengguna, dalam kasus ini *administrator*.

BAB VI

PENUTUP

Bab ini membahas mengenai kesimpulan yang diambil dari berdasarkan perancangan perangkat lunak, serta hasil uji coba yang telah dilakukan. Selain itu, terdapat beberapa saran untuk pengembangan lebih lanjut dari aplikasi ini.

6.1 Kesimpulan

Dari hasil pengamatan selama perancangan, implementasi, dan proses uji coba perangkat lunak yang dilakukan, penulis mengambil kesimpulan sebagai berikut :

1. Dari hasil uji coba dapat disimpulkan bahwa sistem telah menjalankan semua proses dan telah memenuhi kebutuhan utama dalam menerapkan fitur *menu login web*, *menu utama administrator*, *menu peta administrator*, *menu entry administrator*, *menu peta user*, *menu login android*, *menu entry user*, dan *menu peta user*.
2. Dari hasil uji coba fungsionalitas pada sistem pencatatan data lokasi, semua fungsi dan proses-proses telah berjalan sebagaimana mestinya.
3. Dari hasil uji coba performa pada sistem pencatatan data lokasi, *menu login web* mempunyai *time/latency* 83 ms/ 81 ms, *menu utama administrator* mempunyai *time/latency* 64 ms/ 62 ms, *menu utama administrator* mempunyai *time/latency* 23 ms/ 21 ms, *menu entry administrator* mempunyai *time latency* 126 ms/ 124 ms, *menu peta user* mempunyai *time/latency* 49 ms/ 49 ms, *menu login android* mempunyai *time/latency* 495 ms/ 495 ms, dan *menu entry user* mempunyai *time/latency* 185 ms/ 185 ms.

6.2 Saran

Adapun saran-saran yang dapat diberikan untuk pengembangan lebih lanjut mengenai aplikasi ini dapat dijabarkan sebagai berikut :

1. Untuk memperindah tampilan *web* maupun *android*-nya, perlu adanya *re-design*.
2. Untuk mempertajam analisis mengenai sistem ini sebaiknya metode maupun data pengujiannya diperbanyak serta dikembangkan ke lebih banyak contoh kasus.
3. Data lokasi perlu diperbanyak dan disesuaikan dengan kondisi kekinian.

DAFTAR PUSTAKA

- [1] “Ravi Tamada, *Android Login and Registration with PHP, MySQL and SQLite*”. (2012).
[Online]. <http://www.androidhive.info/2012/01/android-login-and-registration-with-php-mysql-and-sqlite>.
Diakses : 25 Februari 2014.
- [2] “Ravi Tamada, *Android Session Management using Shared Preference*”. (2012).
[Online]. <http://www.androidhive.info/2012/08/android-session-management-using-shared-preferences>.
Diakses : 1 Maret 2014.
- [3] “Reference Android API’s”. [Online].
<http://developer.android.com/reference/packages.html>.
Diakses : 5 Maret 2014.
- [4] “Amri Shodiq, *Tutorial Dasar Pemrograman Google Maps API*”.(2009). Diakses : 6 Desember 2013

LAMPIRAN

Bagian ini merupakan lampiran yang berisikan potongan kode implementasi dari *script/code* yang digunakan untuk membangun aplikasi.

A. Check Login Pada Web

Berikut ini merupakan potongan *script* yang menjabarkan bagaimana jalannya proses pengecekan *login* pada *web*.

Kode Implementasi 1 Check Login Pada Web

```
<?php

$host = "localhost"; // Host name
$username = "root"; // Mysql username
$password = ""; // Mysql password
$db_name = "tugas_akhir"; // Database name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password") or
die("koneksi ke database gagal");
mysql_select_db("$db_name") or die("database tidak
ditemukan");

// username and password sent from form
$username = $_POST['username2'];
$password = $_POST['password2'];

// To protect MySQL injection (more detail about MySQL
injection)
$username = stripslashes($username);
$password = stripslashes($password);
$username = mysql_real_escape_string($username);
$password = mysql_real_escape_string($password);

$sql = "SELECT * FROM data_user WHERE id_user='$username'
and password='$password'";
$result = mysql_query($sql);
$data = mysql_fetch_array($result);

if($username == $data['id_user'])
{
    if($password == $data['password'])
    {
        session_start();
```

```

        $_SESSION['username'] = $data['id_user'];
        $_SESSION['password'] = $data['password'];
        $_SESSION['jabatan'] = $data['jabatan'];

        $_SESSION['namadepan'] = $data['nama_depan'];
        $_SESSION['namabelakang'] =
$data['nama_belakang'];

        if($_SESSION['jabatan'] == 'admin')
        {
            header("location:login_admin.php");
        }
        else if($_SESSION['jabatan'] == 'user')
        {
            header("location:login_user.php");
        }
        else
        {
            header("location:login_failed.php");
        }
    }
    else
    {
        header("location:login_failed.php");
    }
}
else
{
    header("location:login_failed.php");
}
?>

```

B. Check Menu Entry Pada Web

Berikut ini merupakan potongan *script* yang menjabarkan bagaimana jalannya proses pengecekan *entry* data lokasi pada *web*.

Kode Implementasi 2 Check Menu Entry Pada Web

```

<?php

$host = "localhost"; // Host name
$username = "root"; // Mysql username
$password = ""; // Mysql password
$db_name = "tugas_akhir"; // Database name

// Connect to server and select databse.

```



```

mysql_connect("$host", "$username", "$password")or
die("koneksi ke database gagal");
mysql_select_db("$db_name")or die("database tidak
ditemukan");

$username = $_POST['namauser'];
$namalokasi = $_POST['namalokasi'];
$namapelanggan = $_POST['namapelanggan'];
$lintang = $_POST['latitude'];
$bujur = $_POST['longitude'];
$keterangan = $_POST['keterangan'];

$sql = "INSERT INTO `data_location`(`id_user`,
`nama_lokasi`, `nama_pelanggan`, `lintang`, `bujur`,
`keterangan`) VALUES ('$username', '$namalokasi',
'$namapelanggan', '$lintang', '$bujur', '$keterangan')";
$result = mysql_query($sql);
header("location:page_admin_entry_success.php");

?>

```

C. JSON Parser pada Android

Berikut ini merupakan potongan *code* yang menjabarkan bagaimana jalannya JSON Parser pada *android*.

Kode Implementasi 3 Class JSON Parser

```

package com.jsonparser;

import org.json.*;
import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;

import java.io.*;
import java.util.List;

public class JsonParser {

    static InputStream is = null;
    static JSONObject jo = null;
    static String result = "";
    public JSONObject getJSONfromUrl(String url,
List<NameValuePair> params)

```

```

        {
            try
            {
                HttpClient httpClient = new
DefaultHttpClient();
                HttpPost httpPost = new HttpPost(url);
                httpPost.setEntity(new
UrlEncodedFormEntity(params));
                HttpResponse response =
httpClient.execute(httpPost);
                HttpEntity entity =
response.getEntity();
                is = entity.getContent();
            }
            catch(Exception e)
            {
                return null;
            }

            // Read response to string
            try
            {
                BufferedReader reader = new
BufferedReader(new InputStreamReader(is,"utf-8"),8);
                StringBuilder sb = new
StringBuilder();
                String line = null;
                while ((line = reader.readLine()) !=
null)
                {
                    sb.append(line + "\n");
                }
                is.close();
                result = sb.toString();
            }
            catch(Exception e)
            {
                return null;
            }

            // Convert string to object
            try
            {
                jo = new JSONObject(result);
            }
            catch(JSONException e)
            {
                return null;
            }
        }
    }

```

```

        return jo;
    }
}

```

D. Shared Preference pada Android

Berikut ini merupakan potongan *code* yang menjabarkan bagaimana jalannya *shared preference* pada *android*.

Kode Implementasi 4 Class Session Manager

```

package com.function;

import com.activity.LoginActivity;

import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.content.SharedPreferences.Editor;

public class SessionManager {

    SharedPreferences sp;
    Editor editor;
    Context _context;

    private static final String nama_sp = "PUSUME";
    private static final String nama_username =
"USERNAME";
    private static final String status_login =
"IS_LOGIN";

    public SessionManager(Context context){
        this._context = context;
        sp = _context.getSharedPreferences(nama_sp, 0);
        editor = sp.edit();
    }

    public void createSession(String username)
    {
        editor.putBoolean(status_login, true);
        editor.putString(nama_username, username);
        editor.commit();
    }

    public String getUsername(String username)
    {
        sp.getString(nama_username, username);
        return username;
    }
}

```

```

    }
    public void removeUsername()
    {
        editor.remove(nama_username);
        editor.commit();
    }

    public void checkLogin(){
        // Check login status
        if(!this.IS_LOGIN())
        {

            // user is not logged in redirect him to Login
Activity
            Intent i = new Intent(_context,
LoginActivity.class);

            // Closing all the Activities from stack
            i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);

            // Add new Flag to start new Activity
            i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);

            // Staring Login Activity
            _context.startActivity(i);
        }
    }
    private boolean IS_LOGIN() {
        // TODO Auto-generated method stub
        return sp.getBoolean(status_login, false);
    }
}

```

E. AsyncTask pada Android

Berikut ini merupakan potongan *code* yang menjabarkan bagaimana jalannya *asyntask* pada *class Login Activity* di *android*.

Kode Implementasi 5 Class AsyncTask

```

package com.activity;

import com.jsonparser.JsonParser;
import com.function.LoginFunction;
import com.function.SessionManager;

import org.json.JSONException;

```

```

import org.json.JSONObject;

import com.activity.R;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Bundle;
import android.sax.StartElementListener;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class LoginActivity extends Activity
{
    private static final String KEY_USERNAME = "id_user";
    private static final String KEY_SUCCESS = "success";

    EditText t_username;
    EditText t_password;
    Button t_login;

    JsonParser jsonparser = new JsonParser();
    SessionManager session;

    Boolean isInternetPresent;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        t_username = (EditText)
findViewById(R.id.editText1);
        t_password = (EditText)
findViewById(R.id.editText2);
        t_login = (Button)
findViewById(R.id.button1);

        t_login.setOnClickListener(new
View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method

```

```

stub
    LoginUser startlogin = new
LoginUser();
    startlogin.execute();
    }
    });

    }

    private class LoginUser extends AsyncTask<String,
Void, JSONObject>
    {
        ProgressDialog dialog;

        @Override
        protected void onPreExecute()
        {
            dialog =
ProgressDialog.show(LoginActivity.this, "Loading...",
"Username and Password diproses...");
        }

        @Override
        protected JSONObject doInBackground(String...
params)
        {
            String username =
t_username.getText().toString();
            String password =
t_password.getText().toString();

            LoginFunction loginfunction = new
LoginFunction();
            JSONObject json =
loginfunction.loginUser(username, password);
            return json;
        }

        @Override
        protected void onPostExecute(JSONObject json)
        {
            dialog.dismiss();

            try
            {
                int success;
                if(json!=null &&

```

```

!json.isNull(KEY_SUCCESS))
{
    success =
json.getInt(KEY_SUCCESS);
    if (success==1)
    {
        JSONObject json_user = json.getJSONObject("user");
        String
username = json_user.getString(KEY_USERNAME);
        session
= new SessionManager(getApplicationContext());
        session.createSession(username);
        String
session_username = session.getUsername(username);
        Log.d("konek", session_username);
        Intent i
= new Intent(getApplicationContext(),
WelcomeActivity.class);
        i.putExtra("SESSION_USERNAME", session_username);
        i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(i);
    }
    else
    {
        //Log.d("Gagal", json.getString(KEY_MESSAGE));
        Toast.makeText(getApplicationContext(), "Login Gagal,
Username/Password Salah...", Toast.LENGTH_LONG).show();
    }
    else
    {
        Toast.makeText(getApplicationContext(), "Login Gagal,
Input Salah...", Toast.LENGTH_LONG).show();
    }
}
}

```

```
        catch(JSONException e)
        {
            e.printStackTrace();
        }
    }
}
```


BIODATA PENULIS



I Putu Sudhyana Mecha, biasa dipanggil Meka, dilahirkan di kota Denpasar, pada tanggal 05 Agustus 1991. Penulis adalah anak pertama dari tiga bersaudara. Pendidikan sekolah dasar ditempuh di SD Percobaan Negeri Tulangampiang Denpasar, pendidikan menengah di SMP Negeri 1 Denpasar, kemudian dilanjutkan di SMA Negeri 3 Denpasar. Tahap sarjana ditempuh di Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya.

Di kampus Teknik Informatika ITS, penulis tercatat sebagai mahasiswa Teknik Informatika angkatan 2009. Di Himpunan Mahasiswa, penulis juga tercatat sebagai anggota Himpunan Mahasiswa Teknik Computer-Informatika (HMTTC).

Selain itu, penulis adalah anggota aktif HMTTC Kabinet Sinergi periode 2010-2011, Staff Ahli Departemen Pengembangan Sumber Daya Mahasiswa (PSDM) HMTTC Kabinet Perjuangan periode 2011-2012, Organizing Committee (OC) Kaderisasi HMTTC 2010-2011, dan Instructor Committee (IC) Kaderisasi HMTTC periode 2012-2013.